REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT DATE	3. DATES COVERED: *(From – To)*
13 February 2008	13 February 2008	10/01/20050-009/30/2

4. TITLE AND SUBTITLE

Transfer Learing and Hierarchical Task Network Representations and Planning

5a. CONTRACT NUMBER
N00173-05-1-G034

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Hector Munoz-Avila

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Lehigh University
526 Brodhead Avenue
Bethlehem, PA 18015

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Naval Resarch Laboratory
4555 Overlook Avenue, SW
Washington, DC

10. SPONSOR/MONITOR'S ACRONYM(S)

AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The goal of this was to contribute with the Naval Research Laboratory i developing Testbed for Integrating and Evaluating Learning Techniques to support the AI researach community by providing a testing platform to AI algorithms. In addition , we want to demonstratet TIELT capabilities i HTN learning tasks

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	C. THIS PAGE		6	Hector Munoz-Avila
					20b. TELEPHONE NUMBER (include area code) 610.758.3797

Transfer Learning and Hierarchical Task Network Representations and Planning

FINAL Report for Grant 541077
from the Department of Computer Science
and Engineering at Lehigh University
Principal Investigador: Héctor Muñoz-Avila

1. Goal

The goal of this was to contribute with the contribute with Naval research Laboratory (NRL) is developing TIELT (**T**estbed for **I**ntegrating and **E**valuating **L**earning **T**echniques) (TIELT, 2005; Aha & Molineaux, 2004) to support the AI research community by providing a testing platform to AI algorithms. In addition, we wanted to demonstrate TIELT capabilities in HTN learning tasks. Hierarchical task network (HTN) planning is an important, frequently studied research topic. Researchers have reported work on its formalisms and applications (Wilkins, 1988; Currie & Tate, 1991; Erol et al., 1994; Smith et al., 1998; Nau *et al.*, 2005). The reason is that HTN planning is the focus of many AI research efforts. In addition, HTN planners are freely available including the SHOP (Nau *et al.*, 1999), which has been downloaded several thousands times since its inception in 1999. Learning HTNs will reduce the effort required to build TIELT's knowledge bases by providing an automated tool that can generate pieces of these knowledge bases..

2. Technical Approach

We developed an API for Call Tom Power 2 (CTP2), which enables an agent to command the AI player in CTP2. CTP2 was developed by Activision. The source code of CTP2 is freely available in the Apolyton web site (please refer to the Apolyton web site for details about CTP2 use). One of Lehigh's tasks in the Transfer Learning Project was to enable the testing of intelligent agents in the turn-based strategy game Call to Power 2. Call to Power 2 was chosen for two reasons. First, it is a game that involves many strategic factors that an agent would have to make decisions on. Some of the many decisions a player or intelligent agent must make include: where to build new cities, whether to explore or expand the civilization, how to dispose forces to attack enemy units, and which technological advances to seek. Second, the source code for Call to Power 2 was made open-source by Activision in 2003, and is maintained by an open-source community called Apolyton [Apolyton, 2007]. This allows us to make any modifications to the game necessary to enable integration with the AI tools used by the Transfer Learning Project.

There were a few difficulties involved with doing this project. First, even though Call to Power 2 has been made open-source, for legal reasons Activision had to remove all documentation from the code. This makes it rather challenging to come to an understanding of the Call to Power 2 system architecture and the design decisions that the

devclopers made along the way. Consequently, it can be very arduous to add additional fcatures to the API as it requires understanding some of the internal functionality of the game in order to interact with it. Additionally, TIELT is a continually evolving system to meet the needs of the Transfer Learning Project. When new versions are released by the Naval Research Lab, the work done to integrate a game with previous versions of TIELT must be converted to the new versions.

A model of the rules of the game world was first defined within TIELT, and the desired actions and percepts an agent can take to play a complete subset of the game were specified. Then the existing Call to Power 2 API was modified and enhanced in order to both allow an outside agent to play a complete subset of the game and to integrate communications with TIELT. The work done for this project will provide an excellent test bed for future experiments in the Transfer Learning Project. Intelligent agents can now be evaluated in conjunction with Call to Power 2.

Here is a list of some of the commands of the API. For as complete list, code source files and demos please visit:

http://www.cse.lehigh.edu/InSyTe/CTP2TieltIntegration/CTP2Integration.html

Call to Power 2 API Functions (A Sample)

Building a city
API call:
bool CityBuild(API_City p_cCity, const API_UnitType p_iUnitType);
Description:
Instruct a city to start building a unit of the specified type. This is only effective if the build queue for the city is empty.
Parameters:
API_City p_cCity // The city to start the production.
API_UnitType p_iUnitType //The type of unit to start producing.
Returns:
Whether or not the city build plans were successfully changed

Creating a city
API call:
bool Settle(API_Army p_aArmy);
Description:
Have an army settle its current location. Obviously, this only works for settler units (or sea equivalents).
Parameters:
API_Army p_aArmy //The army including a settler unit.
Returns:
Whether or not it was possible to settle

Adding city improvements

API call:

bool CityImprove(API_City p_cCity, const API_CityImprovementType p_iImproveType);

Description:

Instruct a city to start building a city improvement of the specified type. This is only effective if the build queue for the city is empty.

Parameters:

API_City p_cCity //The city to start the production.
API_CityImprovementType p_iImproveType //The type of city //improvement to start building.

Returns:

Whether or not the city build plans were successfully changed

In addition to our work on the CTP2 API we developed a new integrated and automated AI planning and learning architecture, called Learn2SHOP. Learn2SHOP departs significantly from the previous works on AI planning and learning in that its modular architecture integrates Hierarchical Task Network (HTN) planning, concept learning, and computer simulations. Using simulations during the planning and learning process enables the system to get information about the outcomes of the actions. We have implemented Learn2SHOP and tested it on a transfer-learning task. The objective of transfer learning is transferring knowledge and skills learned from a wide variety of previous situations to the current, and likely different, previously unencountered problems(s). The experiments with Learn2SHOP have demonstrated the advantages of integrating planning, learning, and simulation in a real-time strategy game engine.

We have tested Learn2SHOP in a transfer-learning task, where the objective was to take knowledge that was acquired under one model and harness it in the learning within another model (e.g., taking lessons that were learned in one game scenario and using them in other game scenarios). The experiments performed by an objective third-party, namely Naval Research Laboratories, demonstrated the effectiveness of our integrated system in a suite of performance measures of knowledge transfer

For further details please refer to the following publications made, in part, with support of this project:

Sanchez Ruiz-Granados, A., Lee-Urban, S. & Munoz-Avila, H., Gonzalez Calero, P. A., Diaz Agudo, B. (2007) Game AI for a Turn-based Strategy Game with Plan Adaptation and Ontology-based retrieval. Proceedings of the ICAPS-07 Workshop on ICAPS 2007 Workshop on Planning in Games. AAAI Press.

Lee-Urban, S., Parker, A., Kuter, U., Munoz-Avila, H., & Nau, D. (2007) Transfer Learning of Hierarchical Task-Network Planning Methods in a Real-Time Strategy Game. Proceedings of the ICAPS-07 Workshop on ICAPS 2007 Workshop on Planning and Learning (AIPL). AAAI Press.